AMENDMENTS TO THE SPECIFICATION:

Please amend the paragraph beginning at page 2, line 3, as follows:

~~**BACKGROUND OF THE INVENTION**~~

Please amend the paragraph beginning at page 2, line 13, as follows:

~~Description of the Prior Art~~

Please amend the paragraph beginning at page 1, line 28, as follows:

In the context of a system in which non-native instructions are translated into native instructions, it often arises that a single non-native instruction may be translated into more than one native instruction. Accordingly, if an interrupt is received during the execution of a sequence of native instructions representing a single non-native instruction, then the non-native instruction may ~~be~~ only partly have been completed, and the state of the processing system may be uncertain. One way of dealing with this would be to provide additional hardware that was triggered upon receipt of an interrupt signal to store the current state of the processing system such that the state could be restored prior to restarting after the interrupt, and so any partially completed non-native instruction would be able to be carried forward to completion. However, such an approach has the disadvantage of incurring an additional hardware overhead, significant additional complexity and may in itself degrade interrupt performance due to the need to save the state of the processing system prior to servicing the interrupt.

Please amend the paragraph beginning at page 4, line 9, as follows:

Viewed from one aspect the present invention provides apparatus for processing data, said apparatus comprising:

(i) a processor core operable to execute operations as specified by instructions of a first instruction set;

(ii) an instruction translator operable to translate instructions of a second instruction set into translator output signals corresponding to instructions of said first instruction set, at least one instruction of said second instruction set specifying an operation to be executed using one or more input variables;

(iii) an interrupt handler responsive to an interrupt signal to interrupt execution of operations corresponding to instructions of said first instruction set after completion of execution of ~~a~~ any currently executing operation; and

(iv) restart logic for restarting execution after said interrupt; wherein

(v) said instruction translator is operable to generate a sequence of one or more sets of translator output signals corresponding to instructions of said first instruction set to represent said at least one instruction of said second instruction set, each sequence being such that no change is made to said one or more input variables until a final operation within said sequence is executed; and

(vi) after occurrence of an interrupt during execution of a sequence of operations representing said at least one instruction of said second instruction set:

(a) if said interrupt occurred prior to starting execution of a final operation in said sequence, then said restart logic restarts execution at a first operation in said sequence; and

(b) if said interrupt occurred after starting execution of a final operation in said sequence, then said restart logic restarts execution at a next instruction following said sequence.

Please amend the paragraph beginning at page 6, line 30, as follows:

Viewed from another aspect the present invention provides a method of processing data, said method comprising the steps of:

(i) executing operations as specified by instructions of a first instruction set;

(ii) translating instructions of a second instruction set into translator output signals corresponding to instructions of said first instruction set, at least one instruction of said second instruction set specifying an operation to be executed using one or more input variables;

(iii) in response to an interrupt signal, interrupting execution of operations corresponding to instructions of said first instruction set after completion of execution of ~~a~~ any currently executing operation; and

(iv) restarting execution after said interrupt; wherein

(v) said step of translating generates a sequence of one or more sets of translator output signals corresponding to instructions of said first instruction set to represent said at least one instruction of said second instruction set, each sequence being such that no change is made to said one or more input variables until a final operation within said sequence is executed; and

(vi) after occurrence of an interrupt during execution of a sequence of operations representing said at least one instruction of said second instruction set:

(a) if said interrupt occurred prior to starting execution of a final operation in said sequence, then restarting execution at a first operation in said sequence; and

(b) if said interrupt occurred after starting execution of a final operation in said sequence, then restarting execution at a next instruction following said sequence.